Classification of Hand Shape with Similar Contour for Sign Language Recognition

Yutaka Yamada, Tadashi Matsuo, Nobutaka Shimada, and Yoshiaki Shirai
Ritsumeikan University, Computer Vision Lab
1-1-1 Noji-higashi Kusatsu, Shiga, Japan
{yamada,matsuo}@i.ci.ritsumei.ac.jp, {shimada,shirai}@ci.ritsumei.ac.jp

Abstract

Recogntion of a sign language image sequence is challengeable because of the variety of hand shapes and hand motions.

This paper proposes a method to classify hand shapes for sign language recognition. To classify hand shapes with similar contour, we use Histogram of Orientated Gradient(HOG). We build binary hand shape classifiers. The classifiers consist of weak classifiers with threshold. To select effective weak classifiers, we use adaboost algorithm.

Experimental result shows effectiveness of our method.

1 Introduction

Sign language is used for communicating to people with hearing difficulties. Recogntion of a sign language image sequence is challengeable because of the variety of hand shapes and hand motions. Sign language recognition from a image sequence requires feature extraction and feature interpretation. Generally features consist of the position, velocity, and other data(ex. facial expression, nodding)[1]. To extract these features from a sign language image sequence, many ways have been proposed.

Okazawa et al proposed optical flow for recognition of sign language words[2]. Lee et al used motion vector to recognize the words[3]. However, these methods don't use hand shape and orientation features of the words.

Kawahigashi et al used hand contour to represent hand shape features[4]. However, their method uses only hand orientation and stretched finger number. These features are not enough to represent hand shape.

To begin with, we extract hand regions by Kawahigashi's method[4]. Histogram of Orientated Gradient(HOG) feature[5] is calculated from the hand region. HOG represents not only edges of contour of hand but also inner edges of hand. This is because HOG is effective to classify shapes with similar contour.

With HOG feature, we make binary classifiers. They are linear combination consisting of weak classifiers. Weak classifiers simply classify 2 classes by threshold.

In the second section, we describe how to extract hand regions. In the third section, we describe how to calculate HOG feature. In the fourth section, we describe classifier. In the fifth section, experiment are described.

2 Feature Extraction

To classify hand shapes, hand regions must be extracted. This section describe how hand regions are extracted.

2.1 Extraction of Hand Regions

Face and hand regions are first extracted using a model of skin color. Since the colors of the face and hands change depending on the environment and the subject, the range of the skin color is determined from the initial image, where the person region is extracted by background subtraction and skin regions are extracted at fixed positions of the person region.

When hands and face regions overlap, they are separated using the context (the previous an the succeeding frames). First, the image of the face and hands just before overlapping are saved as templates. Assuming that those images do not change, hands and face regions are extracted by template matching. This process is repeated until the overlap is terminated. Next, assuming that hand shape changes during overlapping, the hand region is extracted. The image of face and hands just after overlapping are saved as templates and the regions are similarly extracted by template matching. This process is repeated backward to the beginning of the overlapping. Then the timing of the shape change is determined comparing the degree of matching in the forward backward template matching. The process of template matching is illustrated in Figure 1.

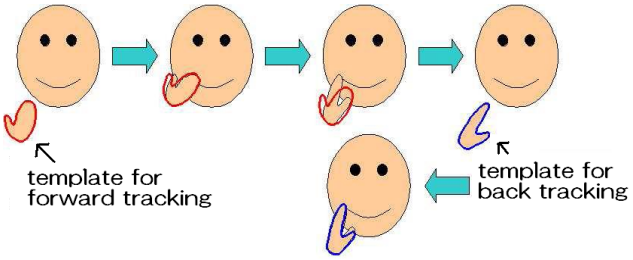


Figure 1. Template Matching

Figure 2. Example of HOG Feature. Left:Input Image, Center:Gradient Image, Right:HOG Image

3 HOG feature

From the hand region extracted in Section 2, we calculate HOG feature. HOG feature was proposed by Dalal et al[5]. It is calculated by counting the occurrences of gradient orientation in locarized portion in an image. The occurrences are repeatedly normalized with some parts overlapped. How to calculate HOG feature is described below.

First,let $L(x, y)$ the luminance of a point of an image. Magnitude of gradient $m(x, y)$ and orientation of it $\theta(x, y)$ is given by,

$$m(x,y) = \sqrt{f_x(x,y)^2 + f_y(x,y)^2} \tag{1}$$

$$\theta(x,y) = \tan^{-1}\frac{f_y(x,y)}{f_x(x,y)} \tag{2}$$

$$f_x(x,y) = L(x+1,y) - L(x-1,y) \tag{3}$$

$$f_y(x,y) = L(x,y+1) - L(x,y-1) \tag{4}$$

Second, from $m(x, y)$ and $\theta(x, y)$, we make an orientation histogram. An input image is divided into $N \times N$ pixel patches. We call this image patch cell. The orientation histogram is calculated for each cell. Each histogram has 9 bins($0^\circ \leq \theta \leq 180^\circ$).

Third process is normalizing process. Each of $M \times M$ cells,this unit is called block, is normalized. When 9 dimension feature of cell(i, j) is expressed as $F_{ij} = (f_1, f_2, ..., f_9)$ and feature vector of k-th block V_k is expressed as $V_k = (F_{ij}, F_{i+1,j}, ..., F_{i+M,j+M})$. Normalized feature v_k is given by,

$$v_k = \frac{V_k}{\sqrt{\|V_k\|^2 + \epsilon}} (\epsilon = 1.0) \tag{5}$$

In normalizing process, cell(i, j) is normalized more than once by different blocks. We define feature vector x as $(v_1, v_2, ..., v_{M^2})$. When N = 5, M = 3, and an input image is 50×50 pixel image, $\mathbf{x}$ has $8block \times 8block \times 81 = 5,184$ dimensions. We use this 5,184 dimension vector to classify hands. An example of Visualization of HOG feature is shown in Figure 2

4 Classifier with HOG Feature

To classify hand shape, we make binary classifiers using HOG features. We make a classifier for each hand shape. Final classifier $H_s(\mathbf{x})$ for hand shape s made by AdaBoost algorithm is expressed as linear combination of T weak classifiers :

$$H_s(\mathbf{x}) = \sum_{t=1}^{T} \alpha_{s,t} g_{s,t}(\mathbf{x}) \tag{6}$$

where $g_{s,t}(\mathbf{x})$ is t th weak classifier for hand shape s and $\alpha_{s,t}$ is a weight of $g_{s,t}(\mathbf{x})$, which is given by

$$\alpha_{s,t} = \ln \frac{1 - \text{err\_rate} + \epsilon}{\text{err\_rate} + \epsilon} \tag{7}$$

where err_rate is weighted error rate of training data. The final result is $sign(H_s(\mathbf{x}))$.

4.1 Generate Classifier

By AdaBoosting algorithm, the final classifier is built. How to make $H_s(\mathbf{x})$ is shown below.
for $t = 1, \cdots$

1. Generate 5,184 weak classifier candidates, corresponding to each dimension of $\mathbf{x}$. $g(i)$, the candidate for i-th value of $\mathbf{x}$ is:

$$g(i) = \text{sign}(x_i - \text{th}_i) \tag{8}$$

   where th_i is a threshold for i-th value. th_i gives the lowest error rate at the interval of 0.01.

2. Calculate evaluate value for each weak classifier. Evaluate value $E_{s,t}(i)$ for $g(i)$ in t th training is:

$$E_{s,t}(i) = \frac{\sum_n^N D_{s,t}(n)(g(i)y_n < 0)}{\sum_n^N D_{s,t}(i)}, \tag{9}$$

   where N is number of training samples, $D_{s,t}(n)$ is the weight of n-th sample for t-th training of shape s, and y_n is the teacher signal of the n-th training sample.

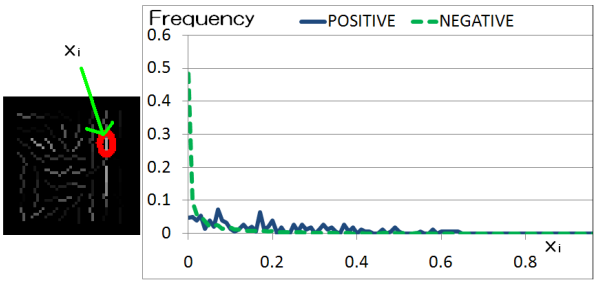


Figure 3. Histogram of x_i of Positive Samples and Negative Samples for Training Images.

3. Define $g_{s,t}(\mathbf{x})$ as $g(i)$, where $i = \arg\min_i E_{s,t}(i)$.
4. Calculate $D_{s,t+1}(n)$. When $H_s(x)$ makes wrong classification for n-th sample, $D_{s,t+1}(i)$ gets bigger than $D_{s,t}(i)$.
5. Calculate error rate of $H_s(\mathbf{x})$. when the error rate is lower than δ, we stop training. δ is experimentally set to 0.0003.

To make $H_s(\mathbf{x})$, we need weight for each training sample. Using y_n,teacher signal of n-th sample, and $D_{s,t}(n)$, weight of i-th sample of (t)-th training for hand shape s, $D_{s,t+1}(n)$ is calculated by

$$D_{s,t+1}(n) = D_{s,t}(n)\exp(-\mathrm{sign}(y_n g_{s,t}(\mathbf{x}))\alpha_{s,t}) \quad (10)$$

Equation 10 makes it possible to select $g_{s,t+1}(x)$ which focuses on samples with false recognition. When $t = 1$, every $D_t(n)$ has the same weight.

In this paper, we use HOG feature as $\mathbf{x}$. If we select appropriate HOG feature x_i and set appropriate th_i, $g_{s,t}(x)$ works well. An example of histogram of x_i for training images is shown in Figure 3. In this case, negative samples are concentrated on from 0.0 to 0.1, so both classes are separable when th_i is set around by 0.1.

5 Experiment

To show effectiveness of our method, we had a classification experiment. We use 28 hand shape classes for the experiment. Figure 4 shows examples of hand shapes. Each classifier for the shape is trained with 100-300 positive samples and 5300-5500 negative samples. Classifiers consist of 20-40 weak classifiers.

We had an experiment of classifying 1500 test images. The test images included words ”ten”, ”silk”, and ”you”, which are difficult to classify by the contour. An image of ”ten” and other similar hand shape images are shown in Figure 5. Each image has a similar contour but belongs to a different class.

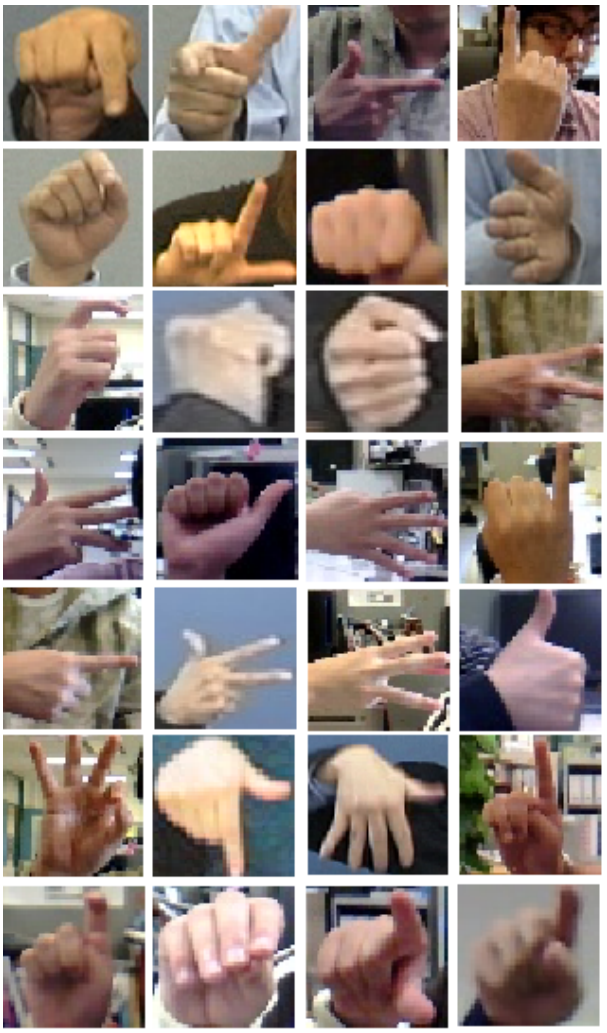


Figure 4. Examples of Each Class

5.1 Result

Experimental result is shown in Table 1. TP in the table represents True Positive. This means only corresponding classfier returns positive output. FP represents False Positive. This means another classifier returns positive output even if corresponding classifier returns positive output. FN represents False Negative. This means no classifier returns positive output. Images with false recognition are shown in Figure 6.

5.2 Discussion

From the Table 1, difference of experimental results between the words is very little. This proves that our method is effective to classify the shapes with similar contour.

We can also say that FN occurs much more often than FP. This is caused by following reasons. One of the biggest

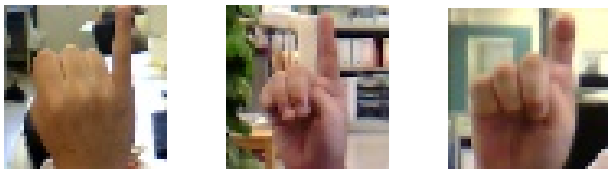


Figure 5. Hand shape with similar contour. Right:"sister",Center:"tomorrow",Left:"ten"

Table 1. recognition rate

input word	TP	FP	FN
ten	0.758	0.040	0.202
silk	0.773	0.042	0.185
you	0.866	0.042	0.091
others	0.759	0.044	0.197

reason is blur and low resolution like the second row of the Figure 6. The other reason is displacement of hand regions in test images. In the third row of Figure 6, Image A is very similar to B and C. These images are a little different in the position of hand. However, this difference makes big difference in HOG feature. Image A is TP and image B and C are FN. To overcome this, we should use more samples in training.

FP occurs in 4% of input images, but in almost all of these cases, correct classifier gives the biggest $H_s(\mathbf{x})$, so it is effective to regard the class with highest $H_s(\mathbf{x})$ as a classification result.

6 Conclusions and Future Work

We proposed a method for classification of hand shape. Classifiers are made of coupling of weak learners with AdaBoost training. From experimental result, our classifiers classify images which have similar contour for other class.

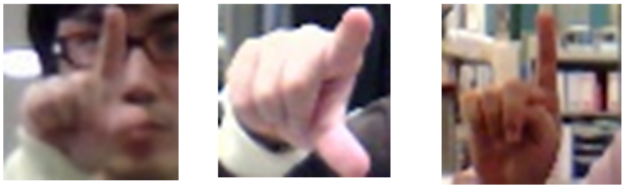
However, our classifier is not good at displacement of hand in an object image. This problem has to be improved.

Now, our method is applied just for the classification of hand shapes, but it can be applied to detect and track hand. We'll make hand detector using these classifiers.

References

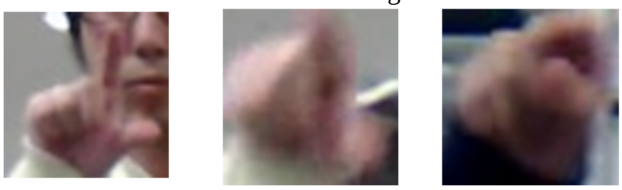
[1] Y.Nagashima, et al,"Computer Processing of Sign Language", the Journal of IEICE, Vol.984, No.4 pp.320-324(2001)

[2] Y.Okazawa et al, "Recognition of Global Motion in Sign Language Based on Optical Flow", Technical Report of IEICE PRMU Vol.104 No.317 pp.39-44(200)



(a)input (b)correct class (c) result

False Positive Images



False Negative Images Caused by Blur and Low Resolution

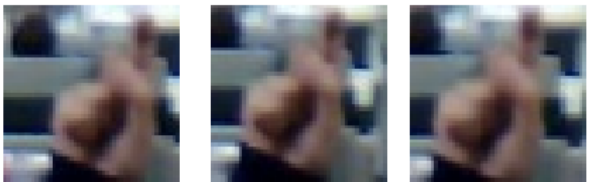


image A image B image C

False Negative Images Caused by Displacement of Hand

Figure 6. False Recognition

[3] M.Lee et al, "Classification of Sign Language using Motion Vector", Technical Report of IEICE , Vol.103 No.746 pp.65-70(2004)

[4] K. Kawahigashi, et al, "Automatic Synthesis of Training Data for Sign Language Recognition Using HMM", Proc. 10th International Conference on Compters Helping People with Special Needs, pp.623-626(2006)

[5] N.Dalal et al,"Histogram of Oriented Gradient for Human Detection", IEEE Computer Vision and Pattern Recognition, Vol.1 pp.886-893(2005)